COMPARISON OF AUTOASSOCIATIVE NEURAL NETWORKS AND KOHONEN MAPS FOR SIGNAL FAILURE DETECTION AND RECONSTRUCTION

THOMAS J. BÖHME
University of Sunderland, School of Computing, Engineering and Technology, Edinburgh Building, Chester Road, SR1 3SD Sunderland, UK

CHRIS S. COX
University of Sunderland, School of Computing, Engineering and Technology, Edinburgh Building, Chester Road, SR1 3SD Sunderland, UK

NICOLAS VALENTIN
CITI - Suez Lyonnaise des Eaux, Technopolis, ZAC de Mercières, 14, rue du Fonds Pernant 60471 COMPIÈGNE, FRANCE

THERRY DENOEUX
Université de Technonologie de Compiègne, UMR CNRS 6599 Heudiasyc, BP 20529, 60205 Compiègne, FRANCE

ABSTRACT
This paper investigates the potential of two different neural network approaches for signal failure detection and reconstruction. Firstly, we propose an approach based on the use of a five-layer perceptron feedforward network (MLP), also known as Autoassociative Neural Network, with a global feedback loop. Two different training methodologies for the Autoassociative neural network are presented in detail. Secondly, we propose an approach based on the use of Kohonen Maps to learn the structure of the data. The map is used for sensor failure detection and different techniques based on simple methods like nearest neighbor are employed to reconstruct the erroneous and missing data. Each missing measurement is estimated from available data in the form of a possibility distribution reflecting the underlying uncertainty. The performance of both neural approaches is illustrated using data collected from a water treatment plant. The two neural networks are trained with the identical training set and identical sensor faults and missing data are imposed. The primary function of the subsequent research is to assess the potential of both neural network types to identify sensor malfunctions and their ability to reconstruct the erroneous sensor signal and missing data. Advantages and disadvantages of the two neural network approaches are discussed.

1. INTRODUCTION

In water treatment, as in many other processes, it is extremely important to provide accurate sensor information. This data is required for many reasons which include legislation, plant efficiency management and process control. Sensor validation is therefore a critical part of overall system operation. Commonly, sensor reliability is assessed using redundant sensor(s) and polling schemes. However, due to the large number of measurements required in the water industry, such techniques are prohibited by cost. Alternative techniques such as observers based upon models of the process are notoriously difficult to implement since even after extensive research many of the unit

operations (especially clarification and filtration) are not well understood (Adgar, 1997). The drive towards "unmanned operation" has also increased demands for greater intelligence, availability and reliability of instrumentation. Therefore, a new generation of "smart" sensors have emerged over recent years. These systems can be programmed to self check and diagnose internal faults, and perform on-line calibration. The aim of this cooperative research was to develop software based methodologies that will improve upon current smart sensor technologies. Two different methodologies are present based upon the use of Autoassociative Neural Networks (AANNs) (Dong, 1994; Guo, 1995; Hines, 1996; Lin, 1991) and Kohonen maps (Kohonen, 1995). The resulting tools enable discrimination between sensor and plant faults; offer process diagnostic advice and provide signal reconstruction support. The “paradigms” are trained from past data alone, information which is extensively available.

This paper is organized as follows. The water treatment operation is first explained in Section 2. The methodology used to build the AANN and the Kohonen map is then described in Section 3 and 4. Finally, experimental results are presented and discussed in Section 5.

2. WATER TREATMENT OPERATION

Water treatment involves physical, chemical and biological processes that transform raw water into drinking water. The Viry-Chatillon water treatment plant, which was used as an application site for this study, provides water to more than 300,000 inhabitants and has a nominal capacity to process 120,000 m^3 of water per day. Figure 1 presents a schematic overview of the various operations necessary to treat the water and the available measurements.Raw water is extracted from the river Seine and pumped to the treatment works. The treatment consists of coagulation-flocculation, settling, filtration, ozonation, filtration and final disinfection. The water is then stored in a tank ready to be transported through the water supply network. The coagulation-flocculation step, which requires the addition of a chemical coagulant, is the critical process to remove colloidal solids.However, contrary to most industrial processes, for which the quality of the input raw material is under control, the quality of the given raw water source may fluctuate due to natural perturbation or occasional pollution. To help overcome these problems, water conditions are monitored upstream of the extraction point at a *Survey Station*. The objective of this paper was to validate the sensor measurements of the survey station as they are the start point in producing optimum treated water quality. Six important measurements with analytical redundancies were selected to make the study: Conductivity, Turbidity, pH, Temperature, Dissolved Oxygen an UV’s absorption.

3. AANN METHODOLOGY

In structural terms an AANN is a five-layer perceptron feedforward network (Kramer, 1991) which can be viewed as two independent three- layer neural networks connected in series as shown in Fig. 2. The first network mixes and compresses the n redundant measurements into a smaller number of characteristic variables which should ideally represent the essential characteristics of the process. The second network works in the opposite way and uses the compressed information to regenerate the original n redundant measurements. The autoassociative neural network consists of 6 inputs and outputs, 4 feature (bottleneck) hidden layer neurons, and 50 neurons for each of the other two hidden layers. The input, output and feature neurons utilize linear activation functions, whereas the two hidden layers for each independent network utilize nonlinear sigmoidal activation functions.

3.1 Sensor Failure Detection and Reconstruction

The detection of sensor failures using the AANN is based on residual generation, produced by comparing the neural network outputs with their corresponding inputs. If the difference between a sensor measurement and its estimate exceeds a specified residual threshold (whilst the other residuals stay relatively low) then that sensor is declared as faulty. In the case that two sensor failures occur simultaneously, both sensors will be seen to have exceeded their thresholds. The thresholds themselves are set up based upon the network error between its sensor inputs and their corresponding estimates under normal operating conditions. Once a faulty sensor measurement is detected, it will be disconnected from the input layer of network to isolate the false information. The external isolating procedure assumes that we can replace the identified faulty sensor input with a reasonably close value to the original one. A good technique for this is to feed the last "good" output value back to the failed sensor input (Böhme, 1999; Böhme, 1998; Guo, 1995), see therefore Fig. 2.

3.2 Data Preparation

The goal of the autoassociative neural network training is to generate good sensor estimates for all sensors even though some of the sensor measurements may have been corrupted.In order to achieve this, the network training also includes the following data pre-processing steps, which can be described as:

- split the data into monthly data sub-sets, in order to improve the reconstruction accuracy,
- mean-center and scale the sensor readings of each month,
- select samples for training, test and final validation of each sub-network,
- colour each sensor variable of the training sets with added noise terms. The noise terms are drawn from a random number generator based on Gaussian distribution with zero mean and standard deviation $\sigma = 1.5 * \varepsilon_i$, where $\pm\varepsilon_i$ is the valid range for the i^{th} sensor reading S_i (Guo, 1991).
- randomize the training data sets,
- Finally, 50% of the training samples of each sensor variable will be coloured to represent corrupted sensor values

After pre- processing is completed select one sub-network to be trained. The back-propagation algorithm is used to adjust the weights of the network so that the network returns the desired sensor measurements for both normal and corrupted sensor measurement conditions. This will allow "isolation" of faulty sensor measurement(s) (Guo, 1995; Guo 1991) from their corresponding input(s) by providing robust output(s) despite the fault(s) in question. Finally, 12 AANN models have been trained each valid for its specific region.

3.3 Training Methodology for the AANN

The training of the autoassociative neural network can be carried out in two ways:

i) training of the entire 5-layer network (Combined Network);
ii) training of the two independent networks separately (Input Training).

3.3.1 <u>Combined Network Training</u>

To train the combined neural network, the weights appearing in the mapping and demapping networks are optimized so that the reconstructed outputs $\mathbf{Z}'$ match the inputs

Z as closely as possible. To do so, the objective function to be minimized is

$$E = \frac{1}{2}\sum_{p=1}^{m}\sum_{k=1}^{n}\left(z_{pk} - z'_{pk}\right)^2 \quad (1)$$

where z_{pk} be the value of the k^{th} observed variable in the p^{th} training sample and z'_{pk} the corresponding AANN approximation.
Consequently, the steepest descent direction for optimizing the network weights $\mathbf{w}_p$ is given by

$$\Delta \mathbf{w}_p = -\frac{\partial E}{\partial \mathbf{w}_p} = \sum_{k=1}^{n}\left(z_{pk} - z'_{pk}\right)\cdot\frac{\partial z'_{pk}}{\partial w_p} \quad (2)$$

For more details about backpropagation see reference (Freeman, 1991).

3.3.2 Input Training

In general, the performance of backpropagation deteriorates as the number of hidden layers gets larger (Hertz, 1991). Thus, instead of training a whole three-hidden layer associative network, one may train its mapping and demapping subnets separately. Training such subnets is meaningful and can be done by extending the backpropagation algorithm, because the error function is well defined. The difference between training the mapping and demapping subnets and training an ordinary feedforward network is that the outputs and inputs (feature values), respectively, to the subnets are not given. To overcome this burden, not only the internal network parameters, but also the input sequence- starting from a good estimate, e.g. linear PCA loadings - will be adjusted to reproduce the given data as accurately as possible (Böhme, 1999**;** Reddy, 1996; Tan, 1995). The steepest descent direction for minimizing the output errors through adjustment of network inputs is derived by minimizing the objective function represent by Eq. (1).
Therefore, the steepest descent direction for optimizing network inputs x_{pi} is given by

$$\Delta x_{pi} = -\frac{\partial E}{\partial x_{pi}} = \sum_{k}\left(z_{pk} - z'_{pk}\right)\cdot\frac{\partial z'_{pk}}{\partial x_{pi}} \quad (3)$$

Assuming that input and output nodes of the demapping network use the identity activation function, while hidden nodes use a sigmoidal function, the steepest descent direction for training network inputs is

$$\Delta x_{pi} = \sum_{j}\upsilon_{ji}\cdot\sigma'\left(b_j + \sum_{i}\upsilon_{ji}\cdot x_{pi}\right)\cdot\sum_{k}w_{kj}\cdot\left(z_{pk} - z'_{pk}\right) \quad (4)$$

where $\sigma(\cdot)$ is a sigmoidal function, b_j is the bias of the j^{th} hidden node, and υ_{ji} and w_{kj} are the network weights of the i^{th} subnet input and k^{th} output.

4. KOHONEN MAP

We propose a second approach based on the use of a Self-Organizing Map (SOM) (Kohonen, 1995) for multi-parameter data validation and reconstruction of data. The SOM is a means for automatically arranging high-dimensional data. The resulting map avails itself readily to visualization, and thus the distance relations between different data items can be illustrated in a familiar and intuitive manner.
The SOM algorithm is a hybrid method in that it combines the goals of projection and clustering algorithms. It can be used at the same time to visualize the clusters in a data set, and to represent the set on a two dimensional map in a manner that preserves the

nonlinear relations of the data items; nearby items are located close to each other on the map. Moreover even if no explicit clusters exist in the map, the SOM methods can reveals "ridges" and "ravines".

4.1 The principle of the SOM

The process in which the SOM is formed is an unsupervised learning process. The SOM defines a mapping from the input data space $\Re^n$ (raw water quality parameters) onto a regular two-dimensional array of nodes (here an hexagonal array) as shown in Fig. 3. A reference vector, or prototype, $m_i \in \Re^n$ is associated to every node i. Each input vector x $\in \Re^n$ is compared with the m_i, and the best match m_c defines the winning prototype. The input is then mapped onto the corresponding location on the grid.

At each step t of the learning process, a data sample $x(t) \in \Re^n$ is presented to the units. The node c that best represents the input is then searched for using, e.g., the Euclidean distance to define the quality of the representation: $\|x - m_c\| = \min_i \{\|x - m_i\|\}$. Next, the unit c as well as neighboring units learn to represent the data sample more accurately. The model vector of unit i is updated according to the following learning rule:

$$m_i(t+1) = m_i(t) + h_{ci}(t)[x(t) - m_i(t)] \quad (5)$$

Here h_{ci} is a 'smearing' or neighborhood function expressing how much the unit i is updates when unit c is the winner. The neighborhood function typically is a symmetric, monotonically decreasing function of the distance of units i and c on the map grid. During repeated application of Eq. (5) with different inputs, model vectors of neighboring map units become gradually similar because of the neighborhood function h_{ci}, eventually leading to global ordering of the model vectors. With time, the m_i then tend to be ordered along with the array in a meaningful way.

4.2 Training Methodology

A map of size 15*15 was trained on a data set consisting of 100,000 measurements of 6 variables (turbidity, conductivity, pH, temperature, dissolved oxygen and UV absorption) taken at 5 min intervals. The SOM-PAK software package (Kohonen et al., 1995) was used for the simulations. Components planes of the map are shown in Fig. 4.

4.3 Sensor Failure detection and reconstruction

Kohonen maps allow not only to visualize the evolution of raw water quality in two dimensions, but also to detect atypical data or outliers by measuring the distance between each input vector and its closest reference vector. More precisely, the activation of unit *i* for input *x* was defined using a Gaussian kernel:

$$K(i) = \exp\left(\frac{-1}{2\sigma_i^2}\|x - m_i\|^2\right) \quad (6)$$

where σ^2_i is a parameter defining the size of the influence region of unit *i*. If the activation of the winning prototype is smaller than a specified threshold, the current sample is considered as invalid. The contributions of each of the components of vector x to the distance $\|x - m_i\|$ are then examined to determine more precisely which sensors should be declared as faulty. These sensor measurements are then disconnected to compute a new winning prototype with only valid parameters. For reconstruction, each missing value of a given input variable is estimated by the value of the corresponding

component in the winning prototype. In order to improve the reconstruction accuracy we use a combination of the k nearest nodes. Each missing or invalid value j is estimated by a combination of the corresponding component in the k nearest prototypes:

$$\hat{x}(j) = \sum_{i=1}^{k} K(i)\, m_i(j) \Big/ \sum_{i=1}^{k} K(i) \qquad (7)$$

where $m_i(j)$ denotes component j of prototype i.

5. COMPARISON RESULT

In order to assess the performance of both sensor validation schemes, an off-line simulation study was performed on the original six sensor variables, capturing one year of data with faults introduced at certain times. In the simulation, each sensor variable was sampled every hour whereas on the physical plant the sampling procedure takes place every 5 min. In the scope of this paper we will consider two scenarios: two cases of single sensor failure and one case when two sensors fail consecutively.

Case 1: *A failure on the process variable pH*
In the first example, the pH sensor is simulated to be degraded with a slow rising ramp of 0.0002/ 5 minutes. The fault occurs on the 18th June 1998 at 12:00. Using the AANN the pH sensor is declared faulty 198 samples (16.5 hours) later on 19th June 04:30 (Figures 5 and 6). Using the Kohonen Map, the pH sensor is declared faulty 117 samples later (9 hours and 45 minutes) later on 18th June 21:45 (Figures 7-9).

Case 2: *A failure on process variable dissolved oxygen*
In the second example, the dissolved oxygen sensor is degraded with a bias of magnitude -0.5 on the 18th June at 12:00.
The AANN declares the dissolved oxygen sensor faulty 9 samples (45 minutes) later (Figures 10 and 11), whereas the Kohonen Map declares the dissolved oxygen sensor faulty immediately at 12:00 (Figures 12-14).

Case 3: *Two consecutive sensor failures with higher fault magnitude*
Firstly, the pH reading suffers from a lost of signal, e.g. damaged measurement or broken wire. The fault occurs on the 18th June 1998 at 12:00. Three days later on the 21st June 1998 at 12:00, the dissolved oxygen sensor is corrupted with a bias of magnitude 2.Using the AANN, the total failure of the pH measurement is immediately recognized. The consecutive fault of the dissolved oxygen sensor is declared one sample later as the fault occurs (Figures 15-17).
Using the Kohonen Map, the total failure of the pH measurement is immediately recognized. The consecutive fault of the dissolved oxygen sensor is also immediately recognized (Figures 18-21).

6. CONCLUSION

In this paper we demonstrate the accuracy of two different ANN based methods for signal failure detection and reconstruction using real data. It is established that both methods are able to reconstruct single soft failure as well as two consecutive faults. Both methods are correlation based and their reconstruction quality becomes poorer if more faults occur consecutively. The methods are based on the correlation between 6 parameters, it is necessary to have at least 50 % valid parameters to reconstruct the signal.

The identification time of a fault using the AANN is slower compared to the Kohonen net because the AANN fault identification algorithm utilizes a combination of the individual sensor residual determination and filtered total sensor residual determination. This procedure is designed to prevent false triggering, but of course slows down the identification time. However, it should be noted that the feedback of the last 'good' value that preserves reconstruction capability is not affected by this delay. The Kohonen net detects immediately a bias in a signal because the activation of the winning prototype decreases immediately. The identification time of a small fault is better with the AANN. Also, the Kohonen net has some problems detecting a very slow rising ramp fault on the sensor. This is due to the fact that we trained 12 AANN's (one per month) and only one Kohonen net (entire year). Comparisons of the two methods are very difficult because of the differences in the training procedures followed, and the types of hardware platform used.

7. FIGURES

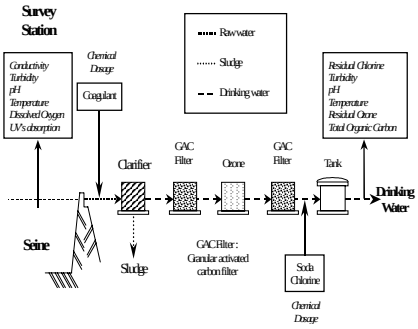

Figure 1

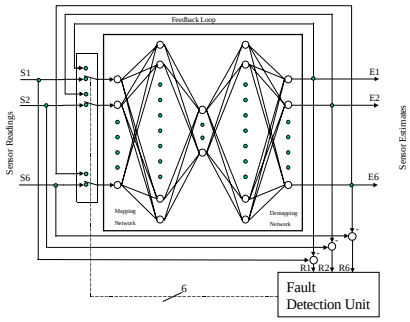

Figure 2

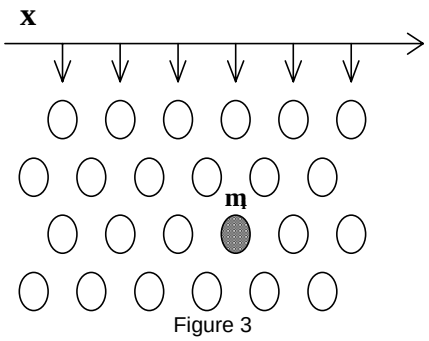

Figure 3

Figure 4

Figure 1: Simplified synopsis of the water treatment plant, Figure 2: Autoassociative Neural Network with feedback loop, Figure 3: Principle of the SOM, Figure 4: Component planes of the map, Figure 5: AANN's reconstruction of pH, Figure 6: AANN's total residual, Figure 7: Activation of the winning prototype, Figure 8: Distance on pH component with the winning prototype, Figure 9: Kohonen map's reconstruction of pH, Figure 10: AANN's reconstruction of Dissolved Oxygen, Figure 11: AANN's total residual, Figure 12: Activation of the winning prototype, Figure 13: Distance on Dissolved Oxygen component with the winning prototype, Figure 14: Kohonen map's reconstruction of Dissolved Oxygen,

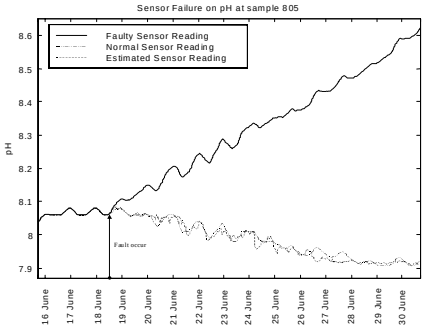

Figure 5

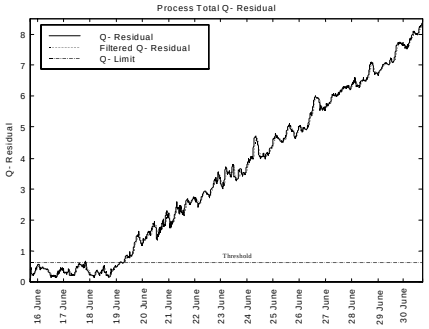

Figure 6

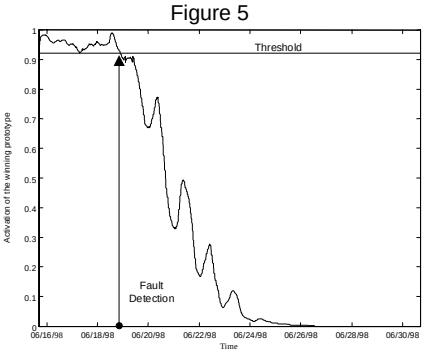

Figure 7

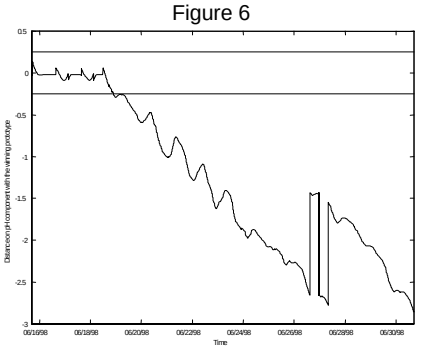

Figure 8

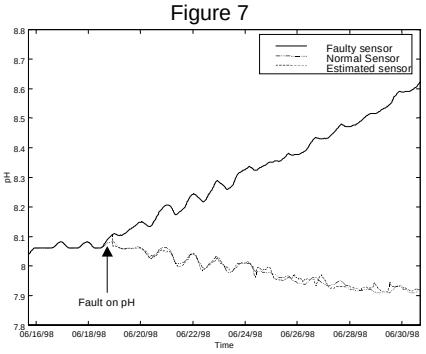

Figure 9

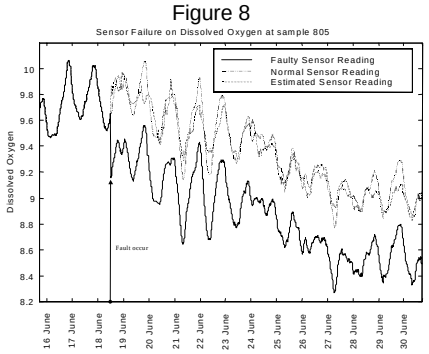

Figure 10

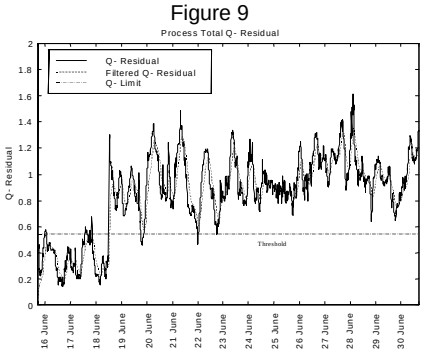

Figure 11

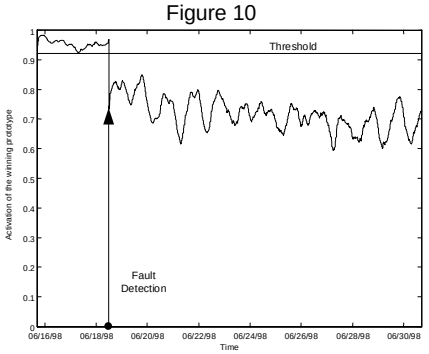

Figure 12

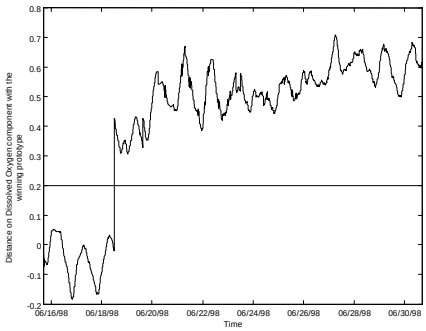

Figure 13

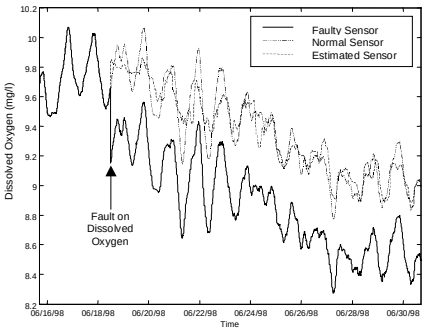

Figure 14

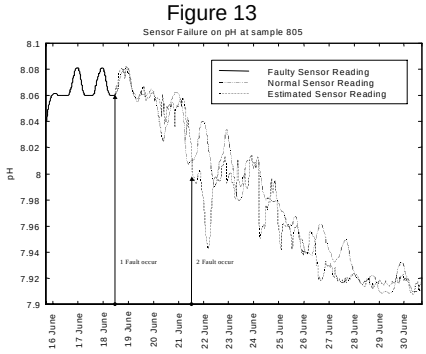

Figure 15

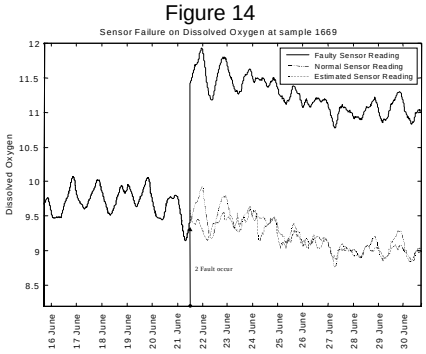

Figure 16

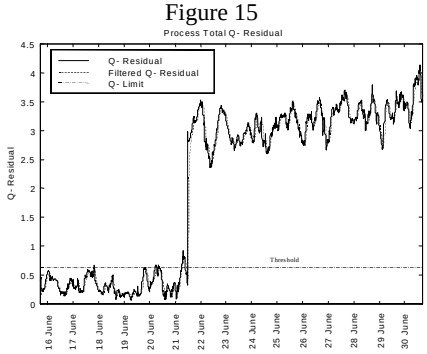

Figure 17

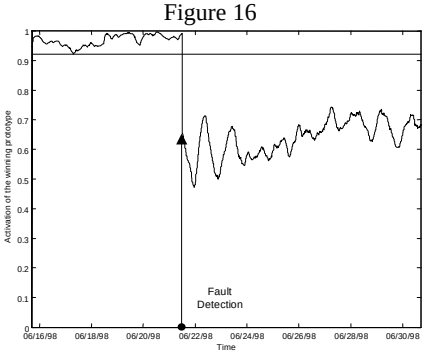

Figure 18

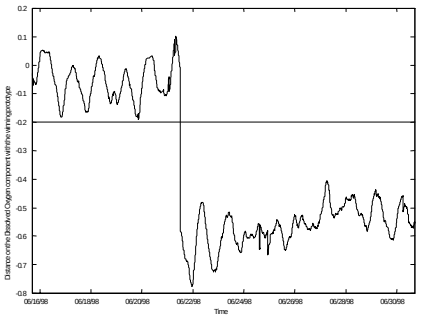

Figure 19

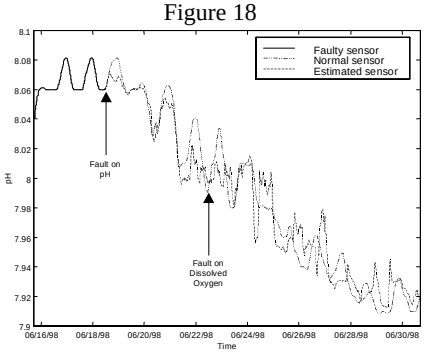

Figure 20

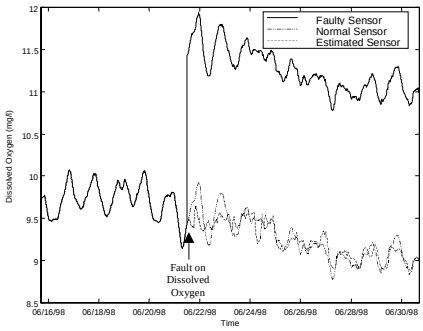

Figure 21

Figure 15: AANN's reconstruction of pH, Figure 16: AANN's reconstruction of Dissolved Oxygen, Figure 17: AANN's total residual, Figure 18: Activation of the winning prototype, Figure 19: Distance on Dissolved Oxygen component with the winning prototype, Figure 20: Kohonen Map's reconstruction of pH, Figure 21: Kohonen map's reconstruction of Dissolved Oxygen

8. REFERENCES

Adgar, A., et al, 1997, "Improving Data Reliability using Statistical and Artificial Network Strategies", *COMADEM'97, 10th International Congress and Exhibition on Condition Monitoring and Diagnostic Engineering Management* - Espoo (Finland), Vol. 2, 9-11 June, 1997.

Böhme, T. J., et al, 1998, "Sensor Failure and Signal Reconstruction using Auto-Associative Neural Networks", *Proc. ICSC/IFAC Symposium on Neural Computation*, pp. 220-225, Vienna, Sept 1998.

Böhme, T. J., et al, 1999, "Reliable Neuro Self-Tuning Control using Autoassociative Neural Networks for the Water Treatment", e&i Elektrotechnik und Informationstechnik ÖDV-Verbandszeitschrift (Austria), special issue: Neural Networks, Vol. 6, pp. 375-389.

Freeman, J.A. and Skapura, D.M, 1991, "*Neural Networks: Algorithm, Applications and Programming Techniques*". Addision-Wesley, New York.

Dong, D. and McAvoy, T., 1994, "Sensor Data Analysis using Autoassociative Neural Nets", *World Congress on Neural Networks* - San Diego' 94, Vol. 1, pp. 161-166, 1994.

Guo, T.-H. and Nurre, J.; 1991, "Sensor Failure Detection and Recovery by Neural Networks", IEEE, pp. 221- 226, 1991.

Guo, T.-H. and Musgrave, J., 1995, "Neural Network based Sensor Validation for Reusable Rocket Engines", *Proc. of the American Control Conference*, Seattle, pp. 1367-1372.

Hertz, J., et al., 1991, "Introduction to the Theory of Neural Computation", Addison-Wesley, Redwood City, CA, p. 198.

Hines, J. W., et al., 1996., "Plant Wide Sensor Calibration Monitoring", *Proc. of the 1996 IEEE International Symposium on Intelligent Control*, Dearborn, MI, pp. 378-383.

Kohonen, T., 1995, "*Self-Organizing Maps*", Heidelberg: Springer Verlag.

Kohonen, T., et al., 1995, "SOM_PAK: The self-organizing map program package", obtainable via anonymous ftp from the internet address "cochlea.hut.fi".

Kramer, M.A., 1991, "Nonlinear Principal Component Analysis using Autoassociative Neural Networks". AIChE Journal, Vol. 37, No. 2, pp. 233-243.

Lin, C.-S., et al., 1991, " Neural Networks for Sensor Failure Detection and Data Recovery", *Proc. of the Artificial Neural Networks in Engineering' 91 Conference*, St. Louis, MO, November, 1991.

Reddy, V. N. and Mavrovouniotis, M. L., 1996, "Plant Monitoring and Diagnosis using Input-Training Neural Networks", *Proc. of the American Power Conference*, Vol. 1, pp. 309-314.

Tan, S. and Mavrovouniotis, M.L., 1995, "Reducing Data Dimensionality through Optimizing Neural Network Inputs", AIChE Journal, Vol. 41, No. 6, pp. 1471-1481.